

REMARKS

Claims 116-162 are pending.

Claims 116-162 have been rejected.

Claims 120, 132, 141, 150, and 159 have been amended. No new matter has been added. Support for this amendment can be found, at least, within the originally-filed specification on page 18, line 5 through page 19, line 20.

Rejection of Claims under 35 U.S.C. § 102

Claims 116-120

Claims 116-162 stand rejected under 35 U.S.C. § 102(e) as purportedly being unpatentable over U.S. Patent No. 6,439,542 (“Koo”). Applicants respectfully traverse this rejection.

Applicants submit that Koo fails to teach or suggest, among other limitations, (1) querying a first and second table from generated SQL statements to produce first and second result sets; and (2) joining the first and second result sets to produce a third result.

Koo presents a method for purportedly optimizing a database query by analyzing the query to identify any joins within the query that are lossless and any tables of the identified joins that are eligible for removal. *See*, Koo Abstract. Koo’s query is then rewritten to eliminate the identified tables that are eligible for removal. *See*, Koo Abstract. Applicants submit that given two tables, Koo’s query optimization does not generate SQL statements that will be performed on the tables, as claimed. Further, Koo does not produce result sets from querying the tables using generated SQL statements, where the result sets are subsequently used in the performance of a join operation, as claimed. Koo’s method is simply as Koo describes it, optimizing a query by identifying tables that can be removed prior to performing the query, and then rewriting the query into a simpler form. Thus, Koo’s query rewriting avoids ever performing any query on the removed tables.

An example of Koo’s method performs a rewrite of the following join query:

```
SELECT COUNT(*)  
FROM STARS ACCOUNT A, STARS.CUSTOMER C  
WHERE A.CUSTID = C.CUSTID AND A.BALANCE < 10
```

Koo 5:5-10. Koo's example join query operates on two tables, a CUSTOMER table, and an ACCOUNT table. Given the above join query, Koo performs an analysis to determine that the CUSTOMER table can be eliminated given the join conditions. *See*, Koo 5:11-22. After eliminating the CUSTOMER table, Koo rewrites the above query as the following:

```
SELECT COUNT(*)
FROM STARS ACCOUNT A
WHERE A.BALANCE < 10
```

Koo 5:22-25. In other words, Koo's system eliminates the need to process one or more tables by rewriting the query without the one or more tables identified for elimination.

Applicants submit that Koo's method begins with an original query and finishes by generating a rewritten query. Generating the rewritten query is Koo's end result; Koo is not executing either the original query or the rewritten query. By contrast, the claimed method begins by receiving two tables, generating SQL statements, querying the two tables to produce two result sets, and finally executing a join operation on the two result sets to produce a third result set. Thus, as an initial matter, Applicants submit that Koo's method is complete before any queries are executed, whereas the claimed method begins by receiving tables and executing queries. In other words, the domains in which Koo's method and the claimed method are relevant do not overlap.

As noted above, Koo's rewriting process does not perform any queries on the CUSTOMER or ACCOUNT tables. Thus, it must be the case that Koo fails to perform queries using any generated SQL statements. At least one reason Koo fails to teach or suggest this limitation is that the CUSTOMER table in Koo is eliminated before either the original query or the rewritten query is performed. As described above, Koo is not performing any queries on the tables, but instead Koo is identifying which tables can be eliminated in order to rewrite the query in a simpler form. Koo's determination of which tables to eliminate does not involve generating a set of SQL statements to query a first and second table, as claimed. Even if Koo could be analogized to the claimed method, at the moment Koo eliminated any table, the analogy would fail because the claimed method performs queries on both the tables it receives.

Further, no part of Koo can be interpreted or characterized as producing result sets from performing any manner of SQL statements on any tables, particularly where such result sets are then used in the performance of a join operation, as claimed. As noted above, because Koo does not perform any queries on the tables that are subject of the original query, it necessarily follows that Koo does not perform any query on the tables using anything comparable to generated SQL statements.

In addition to citing the earlier-presented example in Koo, the Office Action cites to a section of Koo that presents different kinds of joins that can be analyzed and rewritten as purported disclosure of the claimed querying a table to produce a result set, where the query uses a set of generated SQL statements. *See*, Office Action, pp. 3 and 4 (citing Koo 7:10-35). Applicants submit that this section merely details which types of joins can be analyzed using Koo's method. This section of Koo is not an alternative of the above-cited example. Instead, this section presents the details of how the underlying principles of Koo's method, described earlier, operate. Thus, a mere listing of types of joins that can be analyzed by Koo's method in no way teaches or suggests any additional feature, and so is incapable of showing, teaching or suggesting the claimed production of a result set by querying a table with a set of generated SQL statements, by definition.

In addition to the above sections of Koo, the Office Action cites to an example within the "Column Equivalents Predicates" section as purported disclosure of the claimed joining of result sets (result sets produced by performing generated SQL statements on first and second tables). *See*, Office Action, p. 4 (citing Koo 6:40-50). Applicants submit that this section also merely provides the manner in which the earlier-described analysis is performed (these cited sections are within a "Fundamentals" section). This section of Koo does not present an alternative to the earlier-cited example. Instead, this section simply presents aspects of the logical predicates that are used in certain steps of Koo's analysis of the original query, as described earlier. Thus, this section does not describe any sort of alternative to the previously cited section, and so is incapable of showing, teaching, or suggesting any method or aspect different than that already disclosed in Koo. Thus, Koo fails to teach or suggest the claimed joining of first and second result sets in producing a third result set, and in fact, is incapable of so doing.

In summary, the Office Action is citing to two sections of Koo to teach or suggest the above distinctly claimed limitations, when the cited sections teach no more than is already taught in earlier sections of Koo. The cited sections of Koo do not disclose any new or distinct teachings not already disclosed earlier therein. The first cited sections of Koo presents an example of rewriting an original query, while the subsequent sections explain the principles by which the original query is analyzed and rewritten. In other words, these sections of Koo effectively teach no more than the earlier sections of Koo. Clearly, it does not follow that the same teaching can be used as purported disclosure of multiple distinct limitations.

For at least these reasons, Applicants submit that Koo fails to teach or suggest all the elements of claims 116-162. Applicants therefore respectfully request the Examiner's reconsideration and withdrawal of the rejections to these claims.

Dependent Claim 120

Applicants respectfully submit that claim 20 has been amended to recite additional limitations not taught or suggested by Koo. Particularly, Applicants submit that Koo fails to disclose anything comparable to the claimed returning of a result that depends on the original first and second tables received at a first step. This lack of teaching in Koo is evident from the above-cited example in Koo. In the above example, Koo begins with an original query of a CUSTOMER and ACCOUNT table, and after Koo's method is complete, Koo has rewritten the query into a query that has completely eliminated the use of the CUSTOMER table from any and all querying to be performed. Contrary to producing a result that depends on the querying of all tables, as in the claimed method, Koo's method eliminates tables from a query altogether.

For at least these reasons, it cannot be said that Koo's rewritten query result depends on the querying of both the CUSTOMER and ACCOUNT tables. Thus, Applicants respectfully request the Examiner's reconsideration of this claim.

PATENT

CONCLUSION

In light of the amendments and remarks set forth herein, the application and the claims therein are believed to be in condition for allowance without any further examination. Nonetheless, should any issues remain, the Examiner is invited to telephone the undersigned.

If any extensions of time under 37 C.F.R. § 1.136(a) are required in order for this submission to be considered timely, Applicants hereby petition for such extensions. Applicants also hereby authorize that any fees due for such extensions or any other fee associated with this submission, as specified in 37 C.F.R. § 1.16 or § 1.17, be charged to Deposit Account 502306.

Respectfully submitted,

/ Samuel G. Campbell III /

Samuel G. Campbell III
Attorney for Applicants
Reg. No. 42,381
Telephone: (512) 439-5084
Facsimile: (512) 439-5099